

Least Ambiguous Set-Valued Classifiers with Bounded Error Levels

Mauricio Sadinle¹, Jing Lei², and Larry Wasserman^{2*}

¹Department of Statistical Science, Duke University, Durham, NC, USA

²Department of Statistics, Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

In most classification tasks there are observations that are ambiguous and therefore difficult to correctly label. Set-valued classification allows the classifiers to output a set of plausible labels rather than a single label, thereby giving a more appropriate and informative treatment to the labeling of ambiguous instances. We introduce a framework for multiclass set-valued classification, where the classifiers guarantee user-defined levels of coverage or confidence (the probability that the true label is contained in the set) while minimizing the ambiguity (the expected size of the output). We first derive oracle classifiers assuming the true distribution to be known. We show that the oracle classifiers are obtained from level sets of the functions that define the conditional probability of each class. Then we develop estimators with good asymptotic and finite sample properties. The proposed classifiers build on and refine many existing single-label classifiers. The optimal classifier can sometimes output the empty set. We provide two solutions to fix this issue that are suitable for various practical needs.

Keywords: Ambiguous observation; Bayes classifier; Reject option; Multiclass classification; Non-deterministic classifier; Oracle classifier.

*Mauricio Sadinle is a Postdoctoral Associate in the Department of Statistical Science, Duke University, Durham, NC 27708 and the National Institute of Statistical Science — NISS (e-mail: msadinle@stat.duke.edu). Jing Lei is Assistant Professor, Department of Statistics, Carnegie Mellon University, Pittsburgh, PA 15213 (e-mail: jinglei@andrew.cmu.edu). Larry Wasserman is Professor, Department of Statistics and Machine Learning Department, Carnegie Mellon University (e-mail: larry@stat.cmu.edu). The first author was supported by NSF grants SES-11-30706 and SES-11-31897.

1 Introduction

We consider the multiclass classification problem, where the goal is to label each point x in a feature space \mathcal{X} with the appropriate class label y in $\mathcal{Y} = \{1, \dots, K\}$. Unlike standard classifiers, which output a single label for each point in \mathcal{X} , our goal is to construct a set-valued classifier that assigns a set of plausible labels to each point in \mathcal{X} . Our motivation comes from the fact that in most classification tasks there are ambiguous observations whose true class is difficult to determine, yet traditional classifiers are forced to output single labels. We argue that assigning sets of plausible labels provides a better treatment to such instances and therefore leads to a more informative approach to classification.

A set-valued classifier is a function $\mathbf{H} : \mathcal{X} \mapsto 2^{\mathcal{Y}}$, or in other words, $\mathbf{H}(x)$ is a subset of $\{1, \dots, K\}$ for each $x \in \mathcal{X}$. We denote the joint distribution of (X, Y) on $\mathcal{X} \times \mathcal{Y}$ by \mathbb{P} . Having a set-valued output allows us to guarantee levels of confidence in our predictions. We consider two types of coverage guarantees:

$$\begin{array}{ll} \text{Total} & \mathbb{P}\{Y \in \mathbf{H}(X)\} \geq 1 - \alpha, \\ \text{Class-Specific} & \mathbb{P}\{Y \in \mathbf{H}(X) | Y = y\} \geq 1 - \alpha_y \text{ for all } y \in \mathcal{Y}, \end{array}$$

where we refer to α ($\{\alpha_y\}_{y=1}^K$) as the error level(s), and to $1 - \alpha$ ($\{1 - \alpha_y\}_{y=1}^K$) as the coverage or confidence level(s).

Once we fix the desired levels of confidence there are further properties that we want our classifiers to have. In particular, we want a classifier that assigns multiple plausible labels to ambiguous observations, but that does so no more than needed. We therefore would like a classifier \mathbf{H} with minimal *ambiguity*, which we define as

$$\mathbb{A}(\mathbf{H}) = \mathbb{E}\{|\mathbf{H}(X)|\},$$

where $|\cdot|$ is the number of points in a set.

In Section 2 we provide a full characterization of the optimal set-valued classifiers, which we refer to as LABEL (*least ambiguous with bounded error levels*). These optimal classifiers correspond to level sets of the conditional probability functions $p(y|x)$, that is, they have the form $\{y : p(y|x) \geq t_y\}$ for some thresholds $\{t_y\}_{y=1}^K$, where $p(y|x) \equiv \mathbb{P}(Y = y|X = x)$. A potentially undesirable property of the optimal classifiers is that they may lead to empty predictions, that is, $\mathbf{H}(x) = \emptyset$ for some points $x \in \mathcal{X}$, especially when the required coverage is low. We call $\mathcal{N} = \{x : \mathbf{H}(x) = \emptyset\}$ the *null region*. This region arises because minimizing ambiguity can favor making $\mathbf{H}(x)$ empty, and because some classes may be relatively well separated with respect to the coverage requirements. We provide solutions to this issue in Section 3.

We consider generic plug-in estimators in Section 4, together with a technique called *inductive conformal inference* (Papadopoulos et al., 2002; Vovk et al., 2005; Shafer & Vovk, 2008; Vovk, 2013) or *split-conformal inference* (Lei et al., 2014), which we use to adjust the classifiers to have finite sample, distribution-free coverage under essentially no conditions. We will also see in Section 4, that all of our analyses carry through even if we let the number of classes K increase with n as long as $K \equiv K_n = o(\sqrt{n/\log n})$. In Section 5 we present data examples that show the advantages of LABEL classifiers.

1.1 Related work

Classifiers that output possibly more than one label are known as *set-valued classifiers* (Grycko, 1993) or *non-deterministic classifiers* (del Coz et al., 2009). In another related framework called *classification with a reject option* (Chow, 1970; Herbei & Wegkamp, 2006; Bartlett & Wegkamp, 2008; Yuan & Wegkamp, 2010; Ramaswamy

et al., 2015), a classifier may reject to output a definitive class label if the uncertainty is high. Set-valued classification contains this framework as a special case, as one can view the “reject to classify” option as outputting the entire set of possible labels. These methods for set-valued classification generally follow the idea of minimizing a modified loss function. For example, Herbei & Wegkamp (2006) assigns a constant loss $d \in (0, 1/2)$ for the output “reject”, while del Coz et al. (2009) defines the loss function as a weighted combination of precision and recall in an information retrieval framework. Certain components of such modified loss functions, such as the loss of the output “reject” and the weight used to combine precision and recall, lack direct practical meaning and may be hard to choose for practitioners.

Another line of related work is Vovk et al. (2005) and Shafer & Vovk (2008), who introduced a method called “conformal prediction” that yields set-valued classifiers with finite sample confidence guarantees. Lei et al. (2014, 2013), Lei & Wasserman (2014), and Lei (2014) studied the conformal approach from the point of view of statistical optimality in the unsupervised, regression and binary classification cases, respectively. We make use of conformal ideas in Sections 3 and 5. Recently, Denis & Hebiri (2015) used asymptotic plug-in methods to derive classification confidence sets in the binary case. They control a different quantity, namely, the coverage conditional on $\mathbf{H}(X)$ having a single element. Finally, we notice that although it would seem appealing to aim at controlling the conditional coverage $\mathbb{P}\{Y \in \mathbf{H}(X)|X = x\} \geq 1 - \alpha$, for all $x \in \mathcal{X}$, which Vovk (2013) calls “object validity,” Lemma 1 of Lei & Wasserman (2014) unfortunately implies that if X is continuous and $\hat{\mathbf{H}}$ has distribution-free conditional validity, then $\hat{\mathbf{H}}$ is trivial, meaning that $\hat{\mathbf{H}}(x) = \{1, \dots, K\}$.

1.2 Contributions

Our framework improves and generalizes the ideas of [Lei \(2014\)](#) to the case of $K \geq 2$ classes, where K can even grow with the sample size. For binary classification, [Lei \(2014\)](#) proposed to find two prediction regions $C_y \subset \mathcal{X}$, $y = 1, 2$, as the solution to minimizing $\mathbb{P}\{X \in C_1 \cap C_2\}$ subject to $\mathbb{P}\{X \in C_y | Y = y\} \geq 1 - \alpha_y$, $y = 1, 2$, and $C_1 \cup C_2 = \mathcal{X}$. A first difficulty of that approach is that, as stated, the problem cannot be generalized to the multiclass case in a simple manner, and so our extension is technically non-trivial. Most importantly, although [Lei \(2014\)](#)'s construction seems ideal, the interaction of the problem constraints may lead to multiple solutions, some of which do not provide a meaningful treatment of ambiguous observations. If we drop the restriction $C_1 \cup C_2 = \mathcal{X}$, the solution to this optimization problem can correspond to regions that do not naturally overlap, thereby leading to a region of empty predictions (null region). Imposing the constraint $C_1 \cup C_2 = \mathcal{X}$ in such situations leads to multiple solutions, one of which is to fill in the null region with an arbitrary class, which is indeed the solution provided by [Lei \(2014\)](#). That solution, however, conceals the characteristics of the classification task at hand: the null region arises because the classes are relatively well separated with respect to the coverage requirements. In other words, in certain classification tasks we may be able to afford higher confidence levels than the ones initially required. Furthermore, arbitrarily filling in the null region defeats our goal of giving a proper treatment to ambiguous instances, as we illustrate throughout the article, and it is particularly clear in the application to the zip code data in [Section 5.4](#). With multiple classes, arbitrarily filling in the null region no longer corresponds to an optimal solution after imposing the constraint $\cup_{y=1}^K C_y = \mathcal{X}$ (the excess risk of this approach is characterized in [Theorem 7](#)). We therefore provide alternative solutions that give a more appropriate handling of ambiguous instances ([Section 3](#)). Some of our new arguments provide further insights

to the problem and lead to significantly more straightforward characterization and estimation of the optimal classifiers. For example, our Theorems 1 and 5 and lemma 4, and their proofs, are very different from the results presented by [Lei \(2014\)](#).

2 Optimal procedures

Our discussion will focus on the case $\mathcal{X} \subseteq \mathbb{R}^d$. Let \mathbb{P} denote the joint distribution of (X, Y) on $\mathcal{X} \times \mathcal{Y}$. In this section we derive LABEL classifiers assuming that \mathbb{P} is known, but in Section 4 we present different estimation procedures. Let p be the density of \mathbb{P} with respect to $\nu(x, y) = \nu_X(x)\nu_Y(y)$ where ν_X is the Lebesgue measure and ν_Y is the counting measure. Throughout the article, we denote $p(x|y) \equiv p_y(x) \equiv p(x|Y = y)$, where $p(\cdot|Y = y)$ is a density of the conditional distribution of X given $Y = y$, which is assumed to be positive on \mathcal{X} for each $y = 1, \dots, K$. We let $\pi_y \equiv \mathbb{P}(Y = y)$ denote the marginal class probabilities and denote $p(y|x) \equiv \mathbb{P}(Y = y|X = x)$. A set-valued classifier \mathbf{H} can be represented by a collection of sets

$$C_y = \left\{ x \in \mathcal{X} : y \in \mathbf{H}(x) \right\}, \quad \text{for } y = 1, \dots, K.$$

Then, $\mathbf{H}(x) = \{y : x \in C_y\}$. Finally, with a little abuse of notation, we can also define \mathbf{H} as a subset of $\mathcal{X} \times \mathcal{Y}$:

$$\mathbf{H} = \left\{ (x, y) : y \in \mathbf{H}(x) \right\}.$$

Note that $\mathbf{H}(x)$ is the x -section of \mathbf{H} and C_y is the y -section of \mathbf{H} .

2.1 Total coverage

We start by considering the problem of minimizing the ambiguity subject to an upper bound α on the total probability of an error, that is:

$$\min_{\mathbf{H}} \mathbb{E}\{|\mathbf{H}(X)|\} \quad \text{subject to} \quad \mathbb{P}\{Y \notin \mathbf{H}(X)\} \leq \alpha. \quad (1)$$

Theorem 1. *Assume that $p(Y|X)$ does not have a point mass at its α quantile, denoted t_α . The classifier that optimizes (1) is given by*

$$\mathbf{H}_\alpha^* = \left\{ (x, y) : p(y|x) \geq t_\alpha \right\}.$$

This optimal classifier can be written as $\mathbf{H}_\alpha^(x) = \{y : p(y|x) \geq t_\alpha\}$.*

Theorem 1 is a consequence of Lemma 2 by choosing $f = p(x, y)$ and $g = p(x)$. If $p(Y|X)$ has a point mass at its α quantile, define $t_\alpha = \sup\{t : \mathbb{P}\{p(Y|X) \geq t\} \geq 1 - \alpha\}$, and $D_\alpha = \{(x, y) : p(y|x) = t_\alpha\}$. Then we must have $\mathbb{P}(D_\alpha) > 0$. If X has a continuous distribution then we can choose a subset $D' \subseteq D_\alpha$ and let $\mathbf{H}_\alpha^* = \{(x, y) : p(y|x) > t_\alpha\} \cup D'$, with $\mathbb{P}(\mathbf{H}_\alpha^*) = 1 - \alpha$. If X is discrete, such a subset D' may not always exist, but we can use a randomized rule on D_α as in the original Neyman-Pearson Lemma. In the rest of this paper we will avoid this complication by assuming the distribution of $p(Y|X)$ being continuous at t_α .

Lemma 2 (Neyman-Pearson). *Let f and g be two nonnegative measurable functions, then the optimizer of the problem*

$$\min_C \int_C g \quad \text{subject to} \quad \int_C f \geq 1 - \alpha,$$

is given by $C = \{f/g \geq t\}$ if there exists t such that $\int_{f \geq tg} f = 1 - \alpha$.

The problem given in expression (1) is equivalent to the one studied by [Lei \(2014\)](#) for the special case of $K = 2$, but we do not impose the restriction $C_1 \cup C_2 = \mathcal{X}$.

This constraint leads to optimal arbitrary solutions that may not be meaningful for handling ambiguous instances, as argued in Section 1.2. Instead in Section 3 we will provide more principled solutions when the initial classification regions do not cover the whole feature space. Moreover, the characterization provided in Theorem 1 is much simpler than that given in Lei (2014).

Although it seems reasonable to work with procedures that control the total probability of an error, in some circumstances this approach may lead to unsatisfactory classifiers. In particular, when one of the classes is much more prevalent than the others, the probability of properly labeling an element of the smaller classes may be quite low, and it decreases as the probability of the largest class increases. We illustrate this behavior with the following example.

Example 3. Consider $\mathcal{X} = \mathbb{R}$, $\mathcal{Y} = \{1, 2\}$, $\mathbb{P}(Y = 1) = 0.95$, and the distributions $X|Y = y$ being normal with means $\mu_1 = -1$ and $\mu_2 = 1$, and standard deviations equal to 1. In Figure 1 we show the densities of the two classes and the specific coverage of each class as a function of the total coverage. We can see that the probability of correctly labeling an element of class 2 can be quite low, whereas we would correctly label elements of class 1 with probability almost equal to 1.

The previous example indicates that a more appropriate approach should control the coverage of each class, as we show in the next section.

2.2 Class-specific coverage

We now derive LABEL classifiers when controlling the individual coverage of each class. We consider the following problem:

$$\min_{\mathbf{H}} \mathbb{E}\{|\mathbf{H}(X)|\} \quad \text{subject to} \quad \mathbb{P}\{Y \notin \mathbf{H}(X) | Y = y\} \leq \alpha_y \quad \text{for all } y, \quad (2)$$

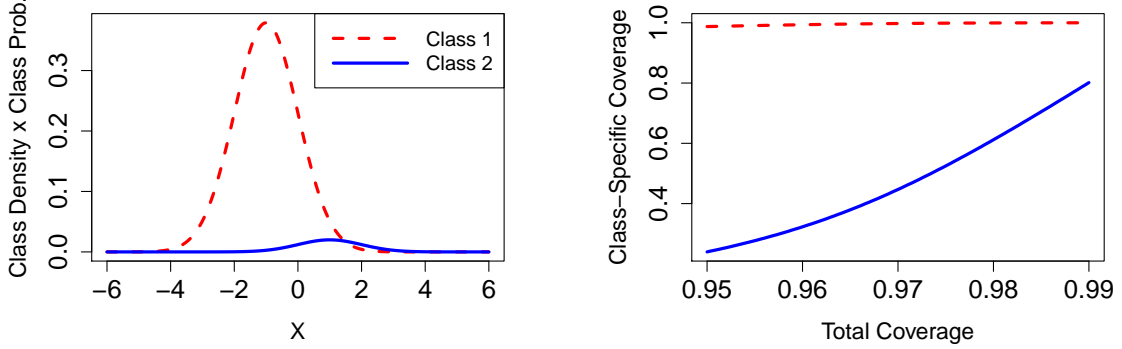


Figure 1: This figure shows the two classes from Example 3. Left: class-specific densities. Right: class-specific coverage as a function of total coverage.

for pre-specified error levels α_y , $y = 1, \dots, K$. Moreover, as we shall see, LABEL classifiers also minimize the probabilities of incorrect label assignments $\mathbb{P}\{y \in \mathbf{H}(X) | Y \neq y\}$ for all y .

Remark 1. *The ambiguity of a set-valued classifier \mathbf{H} can be expressed as*

$$\mathbb{E}\{|\mathbf{H}(X)|\} = \sum_{y=1}^K \mathbb{P}\{y \in \mathbf{H}(X)\}.$$

Lemma 4. *If a set-valued classifier \mathbf{H} minimizes the probabilities of incorrect label assignments $\mathbb{P}\{y \in \mathbf{H}(X) | Y \neq y\}$ for all $y \in \{1, \dots, K\}$ among all classifiers that have certain error levels $\{\alpha_y\}_{y=1}^K$, then it also minimizes the ambiguity.*

Theorem 5. *Given a set of error levels $\{\alpha_y\}_{y=1}^K$, the set-valued classifier induced by the sets $C_y = \{x : p(y|x) \geq t_y\}$, with t_y chosen so that $\mathbb{P}(C_y | Y = y) = 1 - \alpha_y$, simultaneously minimizes the probabilities of incorrect label assignments for all y and the ambiguity.*

Theorem 5 tells us how to find LABEL classifiers when controlling the error probability α_y for each class. The solution may lead to empty predictions, that is, there may exist a region of \mathcal{X} where $\mathbf{H}(x) = \emptyset$. This phenomenon can also occur when controlling

the total probability of an error as in Section 2.1. The presence of this null region occurs when the upper bounds on the error levels are large, when the classes are well separated, or in practice it could happen if we have sample points that are anomalies, that is, points that do not fit any of the existing classes. In any case, we shall propose principled solutions to cover this region, but we first illustrate the procedure given by Theorem 5 with an example and defer the discussion on null regions to Section 3.

Example 6. *We consider an example with $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{1, 2, 3\}$, $\mathbb{P}(Y = y) = 1/3$ for all y , and the distributions $(X|Y = y)$ being bivariate normal with means $\mu_1 = (0, 3.5)$, $\mu_2 = (-2, 0)$ and $\mu_3 = (0, 2)$, and covariance matrices specified as $\Sigma_1 = I_2$, $\Sigma_2 = 2I_2$, and $\Sigma_3 = \text{diag}(5, 1)$, with I_p representing the $p \times p$ identity matrix, and diag representing a diagonal matrix. In Figure 2 we show the classification regions C_y obtained from Theorem 5 for different values of class-specific coverage $1 - \alpha_y$, with $\alpha_y = \alpha$ for all y , $\alpha = 0.2, 0.1, 0.05$. We can see that when the required class-specific coverage is not large enough the procedure can lead to a null region. On the other hand, the null region disappears when the class-specific coverage becomes large and ambiguous classification regions arise as overlaps of the C_y regions.*

3 Dealing with null regions

Given a set-valued classification rule \mathbf{H} , the null region is $\mathcal{N} = \mathcal{N}(\mathbf{H}) = \{x : \mathbf{H}(x) = \emptyset\}$. We present two methods for eliminating these regions of empty predictions.

The first approach, called “*filling with baseline classifier*,” simply uses a fixed baseline classifier to fill in the null region. This method is fast and simple, but it may not properly capture some ambiguous areas of the feature space. Nevertheless, this

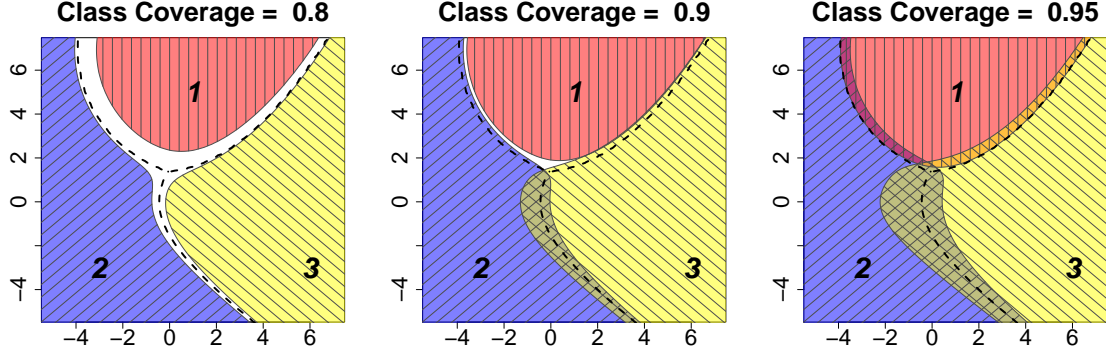


Figure 2: This figure shows the three classes from Example 6. For reference, the black dashed lines denote the boundaries of the Bayes classifier’s regions. Left: Optimal classifier when $\alpha_y = 0.2$ for each class. In this case there is a null region (white) corresponding to $\mathbf{H}(x) = \emptyset$. Middle: Optimal classifier when $\alpha_y = 0.1$ for each class. The null region is smaller and an ambiguous region has appeared. Right: Optimal classifier when $\alpha_y = 0.05$ for each class. The null region is gone but the ambiguity has further increased.

method provides a simple solution if controlling the nominal error levels is the only concern.

The second method, called “*accretive completion*,” gradually grows the optimal classifier by decreasing the class thresholds t_y that define the classes C_y (see Theorem 5), while minimizing the increments in ambiguity, until the null region is eliminated. This approach is more principled and more aligned with the motivation of our framework as it can possibly identify further ambiguous areas inside the null region. We demonstrate this property with the examples presented in Section 5. We recommend this method when thorough detection of ambiguous regions is of concern.

3.1 Approach I: Filling with a baseline classifier

A simple solution to the problem of the null region is to complete the set-valued classifier with a given baseline classifier, such as the Bayes classifier.

Thus, let $h(\cdot)$ be a simple classifier such that $|h(x)| = 1$ for all x , and define

$$\mathbf{H}^\dagger(x) = \begin{cases} \mathbf{H}(x) & \text{if } \mathbf{H}(x) \neq \emptyset \\ h(x) & \text{if } \mathbf{H}(x) = \emptyset. \end{cases}$$

To justify this method, we start from the following optimization problem that explicitly avoids null region:

$$\min_{\mathbf{H}} \mathbb{E}\{|\mathbf{H}(X)|\}, \text{ subject to } \mathbf{H}(x) \neq \emptyset, \forall x, \quad \mathbb{P}(C_y|y) \geq 1 - \alpha_y, \forall y. \quad (3)$$

Problem (3) is hard to solve, in general, but the following theorem says that \mathbf{H}^\dagger is close to optimal when the null region is small.

Theorem 7 (Excess risk bound of \mathbf{H}^\dagger). *Let $\tilde{\mathbb{A}}$ be the optimal value of problem (3), \mathbf{H}^* a solution to (2), and \mathbf{H}^\dagger a classifier such that $|\mathbf{H}^\dagger(x)| = 1$ if $x \in \mathcal{N}(\mathbf{H}^*)$ and $\mathbf{H}^\dagger(x) = \mathbf{H}^*(x)$ if $x \notin \mathcal{N}(\mathbf{H}^*)$ then*

$$\tilde{\mathbb{A}} \leq \mathbb{A}(\mathbf{H}^\dagger) \leq \tilde{\mathbb{A}} + \mathbb{P}\{\mathcal{N}(\mathbf{H}^*)\}.$$

It is important to point out that problem (3) can have multiple solutions, some of which may not necessarily be meaningful when our goal is to properly detect and deal with ambiguous regions, as we explain in Example 8. It is also worth noticing that the excess risk bound of Theorem 7 also characterizes the procedure that fills in the null region with an arbitrary class. This motivates the usage of a different approach, as presented in the next subsection.

Example 8. Consider the scenario given by Example 6 with 0.8 class-specific coverage. In this case the regions C'_y that are the optimal solution of problem (2) are all disjoint and do not cover the whole feature space (left panel of Figure 2). The null region includes cases that are truly ambiguous, that is, cases where either $\mathbb{P}(Y = y|x) \approx 1/3$ for $y = 1, 2, 3$, or $\mathbb{P}(Y = y|x) \approx 1/2$ for two values of Y , and therefore assigning a single label to such cases would not allow us to properly handle their ambiguity. Now, notice that by adding the constraint $\bigcup_{y=1}^K C_y = \mathcal{X}$ to problem (2) we have that the minimum value of the ambiguity is 1. Given that the regions C'_y already achieve the desired levels of coverage, any partition $\{C_y^*\}_{y=1}^K$ of \mathcal{X} such that $C'_y \subseteq C_y^*$ represents an optimal classifier. Interestingly, this includes the option of filling in the null region with an arbitrary class, that is, define $C_y^* = C'_y$ for all $y \neq y_0$ and $C_{y_0}^* = C'_{y_0} \cup \left(\bigcup_{y=1}^K C'_y\right)^c$ for some arbitrary $y_0 \in \{1, \dots, K\}$. We conclude that the problem given in Expression (3) may lead to solutions that are not appropriate for handling ambiguity in classification.

3.2 Approach II: Accretive completion

If a set-valued classifier \mathbf{H} has a non-empty null region $\mathcal{N}(\mathbf{H}) = \{x : \mathbf{H}(x) = \emptyset\}$, we call \mathbf{H} *inadmissible*, otherwise we call \mathbf{H} *admissible*.

Given $\mathbf{t} = (t_1, \dots, t_K)$, denote $\mathbf{H}_{\mathbf{t}} = \{(x, y) : p(y|x) \geq t_y\}$. In Theorem 5 we showed that, for any $\{\alpha_y\}_{y=1}^K$, the solution to the problem (2) is $\mathbf{H}_{\mathbf{t}}$ with \mathbf{t} chosen such that $\mathbb{P}\{Y \in \mathbf{H}_{\mathbf{t}}(X) | Y = y\} = 1 - \alpha_y$. Under this solution the inadmissibility of $\mathbf{H}_{\mathbf{t}}$ implies $\sum_y t_y > 1$. To see this, note that $\mathbf{H}_{\mathbf{t}}(x) = \emptyset$ implies that $p(y|x) < t_y$ for all y . Hence,

$$1 = \sum_y p(y|x) < \sum_y t_y.$$

Therefore, a sufficient condition for \mathbf{H}_t to be admissible is that $\sum_{y=1}^K t_y \leq 1$.

Now suppose that given nominal error levels $\{\alpha_y\}_{y=1}^K$, the solution $\mathbf{H}_{t^{(0)}}$, with $t^{(0)} = (t_1^{(0)}, \dots, t_K^{(0)})$, to problem (2) given by Theorem 5 is inadmissible. We propose to search for a set of thresholds $\{t_y\}_{y=1}^K$ that has the lowest ambiguity and guarantees admissibility as well as nominal coverage of \mathbf{H}_t :

$$\min_{\mathbf{t}} \mathbb{E}\{|\mathbf{H}_t(X)|\} \quad \text{subject to} \quad t_y \leq t_y^{(0)}, \forall y; \quad \sum_y t_y \leq 1.$$

We can proceed in a greedy way to approximate the solution of this problem. The idea is to decrease the thresholds t_y , one at a time, in such a way that at each step we achieve the lowest increment in ambiguity. The detailed procedure is given in Algorithm 1. Notice that for a threshold vector \mathbf{t} the ambiguity function can be written as

$$\mathbb{A}(\mathbf{t}) = \sum_{y=1}^K \mathbb{P}\{p(y|X) \geq t_y\}.$$

Example 9. *To build the intuition for the accretive completion procedure consider the first panel of Figure 3 (nominal coverage 0.95 for each class). We can see that by increasing the coverage of region 1 many points in the null region would be covered by region 1 alone. On the other hand if we wanted to increase the coverage of region 3 we would not cover many points in the null region but we would increase the ambiguity. This indicates that we can approximate the solution to the problem by decreasing the thresholds of the different regions at different rates.*

In Figure 3 the second panel shows the solution given by the accretive completion procedure which leads to ambiguity of 1.028. We can see that the null region was covered mostly by class 1 since t_1 went from 0.99 to 0.206, whereas the other thresholds did not decrease much.

Algorithm 1 The Accretive Completion Algorithm

Require: $\mathbf{t}^{(0)} = (t_1^{(0)}, \dots, t_K^{(0)})$ from the solution of problem (2), step size ϵ

$s \leftarrow 0$

while $\sum_y t_y^{(s)} > 1$ **do**

for $y = 1, \dots, K$ such that $t_y^{(s)} - \epsilon t_y^{(0)} > 0$ **do**

$A_y \leftarrow \mathbb{A}(t_1^{(s)}, \dots, t_y^{(s)} - \epsilon t_y^{(0)}, \dots, t_K^{(s)})$

end for

$y^* \leftarrow \arg \min_{y: t_y^{(s)} - \epsilon t_y^{(0)} > 0} A_y$

$\mathbf{t}^{(s+1)} = (t_1^{(s)}, \dots, t_{y^*}^{(s)} - \epsilon t_{y^*}^{(0)}, \dots, t_K^{(s)})$

$s \leftarrow s + 1$

end while

return $\mathbf{t}^{(s)}$

Although the following remark is obvious from the construction of the accretive completion algorithm, it emphasizes a desirable property of the method.

Remark 2. Let \mathbf{H}^+ be the classifier obtained from the accretive completion procedure and $\{\alpha_y^+\}_{y=1}^K$ its final error levels. Since the algorithm never increases the thresholds t_y , we necessarily have $\alpha_y^+ \leq \alpha_y$, for all y , where α_y is the initial error level. Also, \mathbf{H}^+ is the solution to the problem in expression (2) for error levels $\{\alpha_y^+\}_{y=1}^K$.

It is also possible to use a similar strategy for growing the total coverage classifier. Recall that, in that case, $\mathbf{H}(x) = \{y : p(y|x) \geq t\}$, for a single threshold t . Growing the classifier corresponds to reducing t until there are no null regions. It is easy to see that this happens when $t = 1/K$. In order to have the desired coverage, we also need $t \leq t_\alpha$, where t_α is the threshold given by Theorem 1. We summarize this as a lemma.

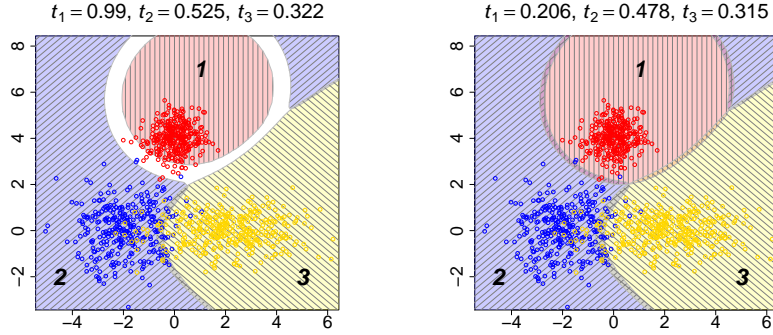


Figure 3: Left: original solution with class coverage of 0.95. Right: solution of accretive completion algorithm, ambiguity 1.028. Sample points are drawn to indicate where the probability mass is concentrated.

Lemma 10. *The classifier of the form $\mathbf{H}_t = \{(x, y) : p(y|x) \geq t\}$ that minimizes the ambiguity, with total coverage at least $1 - \alpha$ and empty null-region is given by $\mathbf{H}(x) = \{y : p(y|x) \geq (1/K) \wedge t_\alpha\}$, where t_α is specified in Theorem 1.*

This lemma suggests that when growing the classifier it is better to use class-specific coverage as above, especially when K is large.

4 Estimation and finite sample adjustment

Now we consider estimating the optimal classifiers using independent draws (X_i, Y_i) , $i = 1, \dots, n$, from \mathbb{P} . We first consider plug-in methods which asymptotically mimic the optimal procedures with rates of convergence under standard regularity conditions. Then we combine these asymptotically optimal procedures with a technique called *split-conformal inference* to achieve a distribution-free, finite sample coverage guarantee.

4.1 The plug-in approach

The optimal classifiers in the previous sections are level sets of $p(y|x)$. There are two things we need to estimate for each $y = 1, \dots, K$: the regression function $p(y|x)$ and the threshold t_y (when controlling total coverage there is a single threshold t_α).

The initial estimator of $p(y|x)$. Any conventional estimator of $p(y|x)$ can be plugged into the optimal classifiers. This could be a direct estimate of $p(y|x)$ or we could estimate π_y and $p_y(x)$ and then use $p(y|x) = \pi_y p_y(x) / \sum_\ell \pi_\ell p_\ell(x)$.

Here are some specific examples:

- (a) *k Nearest Neighbors (kNN)*. For any x , let $d_k(x)$ be the distance from x to its k th nearest neighbor. Define

$$\hat{p}(y|x) = \frac{1}{k} \sum_{i=1}^n I(Y_i = y) I\{\|X_i - x\| \leq d_k(x)\}.$$

We will demonstrate our framework with this estimator in the zip code data example in Section 5.

- (b) *Local polynomial estimator*: $p(y|x)$ is estimated as the regression function of $I(Y = y)$ given $X = x$ using the standard local polynomial estimator ([Tsybakov, 2009](#)). This covers the kernel estimator as a special case:

$$\hat{p}(y|x) = \frac{\sum_{i=1}^n I(Y_i = y) K_h(x - X_i)}{\sum_{i=1}^n K_h(x - X_i)}$$

where K_h is a kernel with bandwidth h .

- (c) *Regularized multinomial logistic regression*:

$$\hat{p}(y|x) = \exp(\hat{\theta}_y^T x) / \{1 + \exp(\hat{\theta}_y^T x)\},$$

where $\widehat{\theta}_y$ is a possibly penalized logistic regression coefficient of $I(Y = y)$ on X .

Estimate the level sets. Having estimated $p(y|x)$ we can determine the cut-off point for the level set according to the target coverage.

Case 1: Total coverage. For the total coverage, define

$$\widehat{\text{Cov}}(t) = \frac{1}{n} \sum_{i=1}^n I\{\widehat{p}(Y_i|X_i) \geq t\}.$$

Let

$$\widehat{t} = \sup\{t : \widehat{\text{Cov}}(t) \geq 1 - \alpha\} = \max_i [\widehat{p}(Y_i|X_i) : \widehat{\text{Cov}}\{\widehat{p}(Y_i|X_i)\} \geq 1 - \alpha].$$

Finally, we take $\widehat{\mathbf{H}}(x) = \{y : \widehat{p}(y|x) \geq \widehat{t}\}$.

Case 2: Class-specific coverage. For the class-specific case, let $\mathbf{t} = (t_1, \dots, t_K)$ and define $\widehat{C}_y = \{x : \widehat{p}(y|x) \geq t_y\}$. The plug-in estimate of the coverage for class y is

$$\widehat{\text{Cov}}_y(t_y) = \frac{\sum_{i=1}^n I(X_i \in \widehat{C}_y) I(Y_i = y)}{\sum_{i=1}^n I(Y_i = y)}.$$

Let

$$\widehat{t}_y = \sup\{t : \widehat{\text{Cov}}_y(t_y) \geq 1 - \alpha_y\} = \max_{i: Y_i=y} [\widehat{p}(Y_i|X_i) : \widehat{\text{Cov}}_y\{\widehat{p}(Y_i|X_i)\} \geq 1 - \alpha_y].$$

Finally, we take $\widehat{\mathbf{H}}(x) = \{y : \widehat{p}(y|x) \geq \widehat{t}_y\}$.

Notice that when $\sum_y \widehat{t}_y > 1$ we can use a plug-in version of the accretive completion (Algorithm 1) to cover the whole feature space. To do this, we replace in Algorithm 1 each $t_y^{(0)}$ by \widehat{t}_y obtained above, and $\mathbb{A}(\mathbf{t})$ by

$$\widehat{\mathbb{A}}(\mathbf{t}) = \frac{1}{n} \sum_{i=1}^n \sum_{y=1}^K I\{\widehat{p}(y|X_i) \geq t_y\}.$$

4.2 Asymptotic properties

There are two main assumptions for the development of asymptotic properties of the plug-in level set estimator. The first one is concerned with the accuracy of \hat{p} . Assume \hat{p} satisfies

$$\mathbb{P} \left\{ \sup_x |\hat{p}(y|x) - p(y|x)| \geq \epsilon \right\} \leq \delta, \quad \forall 1 \leq y \leq K. \quad (4)$$

All conventional estimators mentioned in the previous subsection satisfy this sup-norm consistency under appropriate conditions.

- (a) k NN classifiers: when $p(y|x)$ is Lipschitz on x , and the distribution of X has intrinsic dimension d , then (4) is satisfied by k NN classifiers with $\delta = n^{-1}$ and $\epsilon \asymp (\log n/n)^{1/(2+d)}$ when $d \leq 2$ ($(\log n/n)^{1/(2d)}$ for $d \geq 3$) (Devroye, 1978).
- (b) Local polynomial estimators: when $p(y|x)$ is β -Hölder smooth, and \mathcal{X} has dimension d , then (4) holds with $\epsilon \asymp (\log n/n)^{\beta/(2\beta+d)}$, $\delta \asymp n^{-1}$ for local polynomial estimator of order β (Stone, 1982; Audibert & Tsybakov, 2007; Lei, 2014).
- (c) Logistic regression estimators: in the case where d is large, if the logistic regression model $p(y|x) = \exp(\beta_y^T x) / \{1 + \exp(\beta_y^T x)\}$ holds and $\hat{\beta}_y$ is estimated with appropriately chosen ℓ_1 penalty, then (4) holds with $\epsilon \asymp (\log d/n)^{1/4}$ and $\delta \asymp (d^{-1} + \sqrt{\log d/n} \|\beta_y\|_0)$, provided that minimum eigenvalue of $\mathbb{E}(XX^T)$ is bounded away from 0 (van de Geer, 2008), where $\|\cdot\|_0$ denotes the number of non-zero entries of a vector.

The second assumption is on the density of $p(y|X)$ near the cut-off value. For $1 \leq y \leq K$, let G_y be the distribution function of $p(y|X)$ with X distributed as \mathbb{P}_y , the conditional distribution of X given $Y = y$. Let $t_y = G_y^{-1}(\alpha_y)$ be the ideal cut-off value for $p(y|x)$. Let $G = \sum_{y=1}^K \pi_y G_y$ be the distribution of $p(Y|X)$, with (X, Y)

distributed as \mathbb{P} .

The density condition is, for some constants γ, c_1, c_2, s_0 ,

$$c_1|s|^\gamma \leq |G_y(t_y + s) - G_y(t_y)| \leq c_2|s|^\gamma, \quad \forall s \in [-s_0, s_0], \quad 1 \leq y \leq K. \quad (5)$$

The corresponding condition for the total coverage is

$$c_1|s|^\gamma \leq |G_y(t + s) - G_y(t)| \leq c_2|s|^\gamma, \quad \forall s \in [-s_0, s_0], \quad 1 \leq y \leq K. \quad (6)$$

The difference is that the threshold t is common for all classes.

Theorem 11. *Suppose that (4) and (5) hold, then there exists a constant c such that with probability $1 - K\delta - n^{-1}$ the plug-in class-specific classifier $\{\hat{C}_y\}_{y=1}^K$ satisfies*

$$\mathbb{P}_y \left(\hat{C}_y \triangle C_y^* \right) \leq c \left(\epsilon^\gamma + \sqrt{\frac{\log n}{n}} \right),$$

where \mathbb{P}_y is the conditional distribution of X given $Y = y$. If (6) holds instead of (5), then there exists a constant c such that with probability $1 - K\delta - n^{-1}$ the total coverage classifier $\hat{\mathbf{H}}$ satisfies

$$\mathbb{P} \left(\hat{\mathbf{H}} \triangle \mathbf{H}^* \right) \leq c \left(\epsilon^\gamma + K \sqrt{\frac{\log n}{n}} \right),$$

where \mathbb{P} is the joint distribution of (X, Y) , and \mathbf{H}^* is the corresponding oracle classifier.

Remark: Suppose we let $K \equiv K_n$ increase with n . Then $\mathbb{P}(\hat{\mathbf{H}} \triangle \mathbf{H}^*)$ will still go to 0 as long as $K_n = o(\sqrt{n/\log n})$. Thus, our results include the case where the number of classes increases with n as long as it does not increase too fast.

4.3 Finite sample coverage via split-conformal inference

Using a method called *split-conformal inference* (Lei et al., 2014) (also known as *inductive conformal inference* in Papadopoulos et al., 2002; Vovk, 2013), we can guarantee distribution-free finite sample validity of the classifiers.

Total coverage. The method splits the data in two halves indexed by \mathcal{I}_1 and \mathcal{I}_2 . The first half is used to estimate the conditional probabilities $\widehat{p}(y|x)$, and the second half is used to estimate the distribution of $\widehat{p}(Y|X)$, $(X, Y) \sim \mathbb{P}$. Consider augmenting the second half with a hypothetical sample point (X^*, Y^*) . Under the assumption that this new point is drawn independently from \mathbb{P} , the values $\{\widehat{p}(Y^*|X^*)\} \cup \{\widehat{p}(Y_i|X_i)\}_{i \in \mathcal{I}_1}$ are exchangeable, and so

$$\sigma(X^*, Y^*) \equiv \frac{1}{|\mathcal{I}_2| + 1} \left[\sum_{j \in \mathcal{I}_2} I\{\widehat{p}(Y_j|X_j) \leq \widehat{p}(Y^*|X^*)\} + 1 \right]$$

is uniformly distributed over $\{1/(|\mathcal{I}_2| + 1), 2/(|\mathcal{I}_2| + 1), \dots, 1\}$. Therefore, $\sigma(X^*, Y^*)$ can be used to test the hypothesis $H_0 : (X^*, Y^*) \sim \mathbb{P}$. Given that under H_0 , $\mathbb{P}_\sigma\{\sigma(X^*, Y^*) \leq \alpha\} \leq \alpha$, for $\alpha \in [0, 1]$, we have $\mathbb{P}_\sigma\{\sigma(X^*, Y^*) > \alpha\} \geq 1 - \alpha$, and hence any realization of (X^*, Y^*) such that

$$\sum_{j \in \mathcal{I}_2} I\{\widehat{p}(Y_j|X_j) \leq \widehat{p}(Y^*|X^*)\} > \alpha(|\mathcal{I}_2| + 1) - 1,$$

would not be rejected as being drawn from \mathbb{P} . We then need to find

$$\widehat{t} = \min_{i \in \mathcal{I}_2} \left\{ \widehat{p}(Y_i|X_i) : \sum_{j \in \mathcal{I}_2} I\{\widehat{p}(Y_j|X_j) \leq \widehat{p}(Y_i|X_i)\} > (|\mathcal{I}_2| + 1)\alpha - 1 \right\},$$

and the set of values in $\mathcal{X} \times \mathcal{Y}$ that would not be rejected is given by $\{(x, y) : \widehat{p}(y|x) \geq \widehat{t}\} \equiv \widehat{\mathbf{H}}$. It is thus clear that for any distribution, and any n , $\mathbb{P}_*\{Y \in \widehat{\mathbf{H}}(X)\} \geq 1 - \alpha$, where \mathbb{P}_* is the distribution of the augmented second half of the sample.

Class-specific coverage. To guarantee finite sample, distribution-free validity for class-specific coverage we need to apply the previous method separately to each class. More specifically, as before, we split the data in two halves indexed by \mathcal{I}_1 and \mathcal{I}_2 , and we use the first half to estimate the conditional probabilities $\hat{p}(y|x)$. We partition the second half into K groups corresponding to each class, indexed by $\mathcal{I}_{2,y} = \{i \in \mathcal{I}_2 : Y_i = y\}$, $y \in 1, \dots, K$. Consider augmenting $\{X_i\}_{i \in \mathcal{I}_{2,y}}$ with a hypothetical X^* , under the hypothesis $H_0 : X^* \sim \mathbb{P}_y$. If we follow analogous arguments as for total coverage, we obtain that if we find

$$\hat{t}_y = \min_{i \in \mathcal{I}_{2,y}} \left\{ \hat{p}(y|X_i) : \sum_{j \in \mathcal{I}_{2,y}} I\{\hat{p}(y|X_j) \leq \hat{p}(y|X_i)\} > (|\mathcal{I}_{2,y}| + 1)\alpha_y - 1 \right\}, \quad (7)$$

then the classifier $\hat{\mathbf{H}}$ obtained from the sets $\hat{C}_y = \{x : \hat{p}(y|x) \geq \hat{t}_y\}$ has class-specific finite sample coverage $\mathbb{P}_*\{Y \in \hat{\mathbf{H}}(X) | Y = y\} \geq 1 - \alpha_y$, $y = 1, \dots, K$, where $\mathbb{P}_*(\cdot | Y = y)$ represents the joint distribution of the augmented $\{X_i\}_{i \in \mathcal{I}_{2,y}}$ sample. [Vovk \(2013\)](#) calls this guarantee “label validity.”

The accretive completion (Algorithm 1) can be used easily with the split-conformal estimator because one can just apply it to the second half of the data, while taking $\hat{p}(y|x)$ as an external input. To do this, we replace in Algorithm 1 each $t_y^{(0)}$ by \hat{t}_y obtained from (7), and $\mathbb{A}(\mathbf{t})$ by

$$\hat{\mathbb{A}}(\mathbf{t}) = |\mathcal{I}_2|^{-1} \sum_{i \in \mathcal{I}_2} \sum_{y=1}^K I\{\hat{p}(y|X_i) \geq t_y\}.$$

The theoretical results developed in Theorem 11 also apply to the split-conformal estimator, because the key technical ingredients in the assumption and the proof, such as the sup norm consistency of \hat{p} and the empirical distribution of $p(y|X)$, apply to the split-conformal case.

Theorem 12. *Let $\{\hat{C}_y\}_{y=1}^K$ be the split-conformal classifier constructed from a plug-in*

classifier $\hat{p}(y|x)$. Then the results of Theorem 11 hold for $\{\hat{C}_y\}_{y=1}^K$ under the same conditions, and the classifier has correct distribution-free finite sample coverage.

5 Examples

We now present several examples that illustrate LABEL classification.

5.1 A synthesized nonparametric example

We first illustrate the methods with a nonparametric classifier applied to a simulated two-dimensional dataset of size $n = 1000$ obtained from the distribution presented in Example 9. We estimate $p(y|x)$ using $\hat{p}(y|x) = \hat{p}_y(x)\hat{\pi}_y / \sum_l \hat{p}_l(x)\hat{\pi}_l$, where $\hat{\pi}_y = \sum_i I(Y_i = y)/n$ and $\hat{p}_y(x)$ is a kernel density estimator with bandwidth chosen by Silverman’s rule (Silverman, 1986). We apply the split-conformal prediction approach as described in Section 4.3.

In the top left plot of Figure 4 we show the classification regions \hat{C}_y ($y = 1, 2, 3$) fixing the total coverage at 90%. As we had previously argued, controlling the total coverage can lead to unbalance in terms of the specific coverage of each class. In this case the classes 1, 2, and 3 have estimated specific coverage of 99.4%, 86% and 83.8%, respectively. As an alternative we can control the specific coverage of each class. In the top row and second column of Figure 4 we present the classification regions when fixing the class-specific coverage at 90%. Although this approach guarantees the desired coverage for each class, large null regions tend to appear if we do not require large values of coverage. In the top row and third column of Figure 4 we show the prediction regions when fixing the class-specific coverage at 99%. This eliminates most

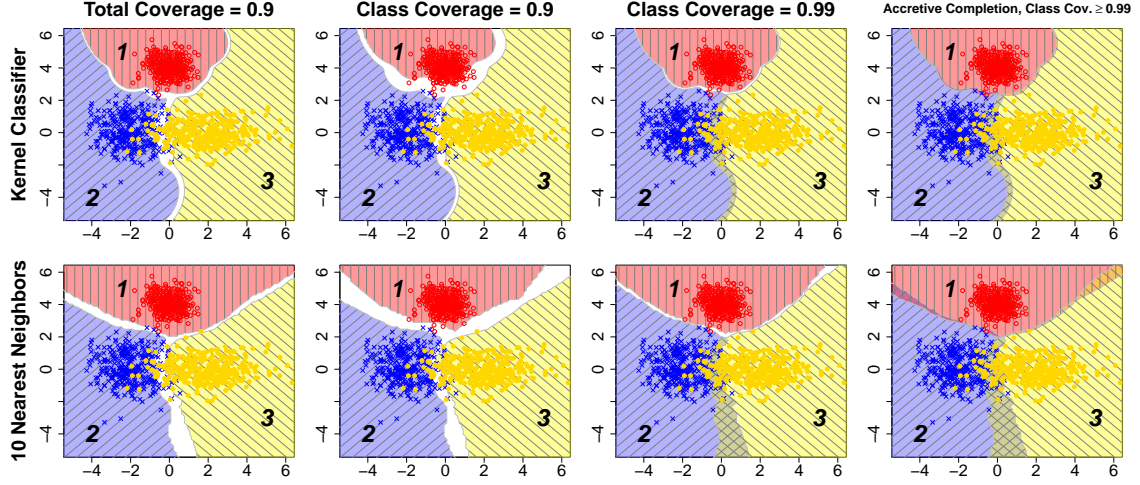


Figure 4: Prediction regions under different coverage conditions (columns) using Kernel and 10-nearest neighbors classifiers (rows). The true data labels are presented as \circ for $Y = 1$, \times for $Y = 2$, and \bullet for $Y = 3$.

of the null region at the expense of growing the ambiguity considerably. Nevertheless, despite the large desired coverage we still obtain a null region due to the fact that class 1 (\circ) is very well separated, and therefore we use the accretive completion algorithm introduced in Section 3.2 to expand the prediction regions, which are shown in the top right plot of Figure 4. The ambiguity of the final classifier is 1.078.

We also explored the performance of a 10-nearest neighbors (10NN) classifier and in the bottom row of Figure 4 we present plots analogous as for the LABEL-kernel classifier. We can see that the classification regions under both approaches are similar in high density areas but they are dramatically different in low density areas. The ambiguity of the final LABEL-10NN classifier is 1.092.

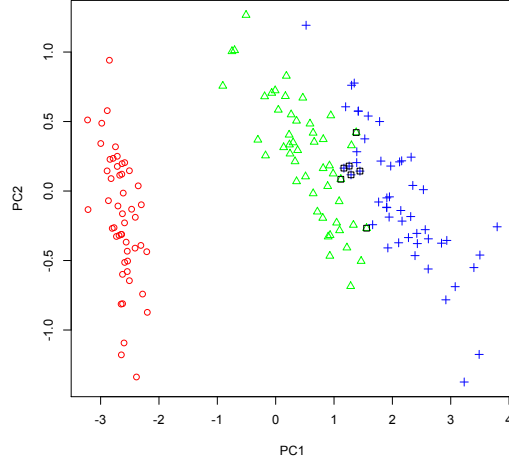


Figure 5: Visualization of the Iris data on the first two principal components. \square indicates ambiguous instances.

5.2 Iris data

We now analyze the well-known Iris data, which consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). There are four features: the length and the width of the sepals and petals, in centimeters. Due to the small sample size, we use these data to illustrate the in-sample behavior of our method. We use a standard multinomial logistic model as the fitting method. To get the set-valued classifier, we first apply our method with class-specific target coverage level 0.95. The resulting classifier has no ambiguity. That is, $\hat{\mathbf{H}}(x)$ is always a singleton for all x in the sample. The in-sample class-specific coverage is $(1, 0.98, 0.98)$, with one error in each of classes 2 and 3. When we increase the target coverage to 0.97, the in-sample coverage is 100%, at a cost of having seven ambiguous instances (three in class 2 and four in class 3) with $\hat{\mathbf{H}}(x) = \{2, 3\}$. In Figure 5, we see that these ambiguous instances are indeed on the boundary between classes 2 and 3.

Table 1: Abalone data: label co-occurrence matrix.

	“1”	“2”	“3”
“1”	329	106	0
“2”	106	478	227
“3”	0	227	361

5.3 Abalone data

The abalone data, available from the UCI Machine Learning Repository ([Lichman, 2013](#)), contains measurements of 4177 abalones. The inference and learning task is to predict the age from eight other easy-to-obtain measurements including sex, length, diameter, height, weight (whole, meat, gut, shell). To ease the presentation, we grouped the age variable into three categories: 1, young (age 0 to 8); 2, middle (age 9 to 10); 3, old (age 11 and older). We randomly split the data into a training sample of size 3342 and a testing sample of size 835.

The estimation starts from the standard multinomial logistic regression with a lasso penalty tuned by the default ten-fold cross-validation. The data exhibits substantial overlap between classes, and we used a class-specific target coverage level 0.8. The test sample class-specific coverage is (0.830, 0.804, 0.781), with ambiguity 1.40. The output classifier has no null region due to the heavy overlap in the training data. The co-occurrence matrix, whose (k, l) entry contains the number of samples classified with labels k and l , is given in Table 1. The classifier does not mix young and old abalones but has difficulty distinguishing between young and middle ones, and between middle and old ones.

Table 2: Zip code data: frequency of prediction sets and ambiguity in test sample.

	$ \widehat{\mathbf{H}}(X) $				$\widehat{\mathbb{E}}\{ \widehat{\mathbf{H}}(X) \}$
	1	2	3	≥ 4	
Test sample frequency	1918	87	2	0	1.045

5.4 Zip code data

Here we apply our method to the zip code data (Le Cun et al., 1990), where the training sample contains 7291 gray scale 16×16 images of hand-written digits, and the class labels correspond to one of the ten digits from “0” to “9”. The class labels are relatively balanced, with the most frequent digit “0” having a proportion of 16.8% and the least frequent digit “8” having 7.6%. The test sample has 2007 images. The zip code data has been analyzed using a similar framework in Lei (2014), but there the problem is converted to a binary one in an *ad hoc* manner. Here we treat it much more naturally as a multiclass problem and reveal some interesting features.

To start from a standard classifier, we use a simple k NN classifier with $k = 10$ chosen by standard three fold cross-validation on the training sample. Then we generate the set-valued classifier with target class-specific coverage level 0.95. We use the split-conformal method described in Section 4.3, using two thirds of the training sample to fit the classifier, and the remaining one third to find the thresholds \widehat{t}_y . There are a few instances in the test sample such that $\widehat{\mathbf{H}}(x) = \emptyset$, we firstly filled these instances by setting $\widehat{\mathbf{H}}(x) = \{\widehat{y}_{\text{knn}}(x)\}$, the simple 10NN classifier.

From Tables 2 and 3 we see that the classifier gives good test sample coverage with small ambiguity. The co-occurrence matrix reported in Table 3 indicates that

- Digits “3”, “5”, “8” are hard to tell from each other;

Table 3: Summary results of applying the LABEL classifier to the zip code data.

Label Co-occurrence Matrix											Class Coverage
	"0"	"1"	"2"	"3"	"4"	"5"	"6"	"7"	"8"	"9"	
"0"	324	0	0	0	0	0	0	0	1	0	0.972
"1"	0	277	0	0	0	0	0	0	0	0	1.000
"2"	0	0	208	1	4	4	1	7	14	0	0.940
"3"	0	0	1	198	0	28	0	1	22	0	0.957
"4"	0	0	4	0	185	1	0	0	6	14	0.932
"5"	0	0	4	28	1	189	3	2	32	3	0.961
"6"	0	0	1	0	0	3	171	0	4	0	0.988
"7"	0	0	7	1	0	2	0	195	6	11	0.937
"8"	1	0	14	22	6	32	4	6	214	5	0.930
"9"	0	0	0	0	14	3	0	11	5	210	0.989

- Digits "0" and "1" are easy to classify;
- Digit "7" is also easy, but occasionally tends to be classified as "2".

We also illustrate the accretive completion method, which can detect additional ambiguous cases in the null region. The top row of Figure 6 gives some typical examples of the ambiguous images in the testing sample without applying the accretive completion algorithm. The bottom row of Figure 6 gives examples of images that are in the original null region but reported as ambiguous by accretive completion. The price to pay for eliminating the null predictions is to increase the ambiguity slightly, but we can see that many of the images that received multiple labels after accretive completion are truly ambiguous, even to the human eye, indicating that filling-in the null region with single label assignments can be potentially misleading.

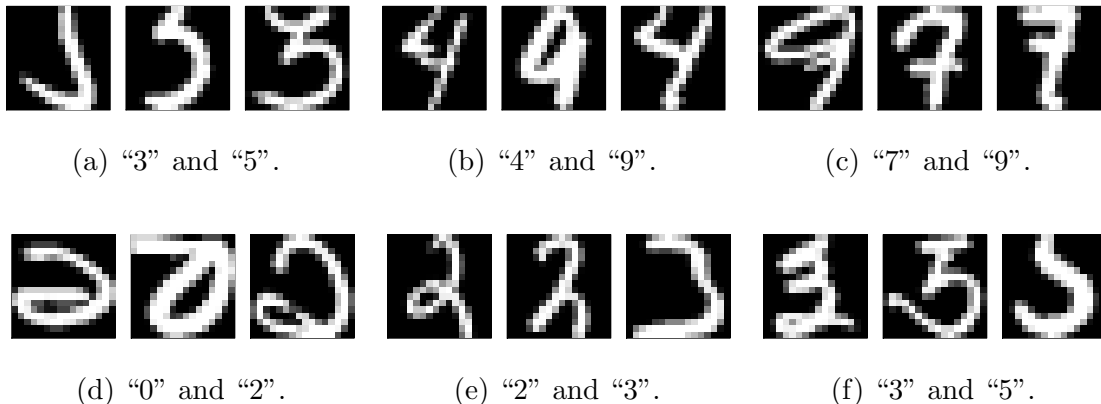


Figure 6: Panels (a)–(c): ambiguous images in initial classifier. Panels (d)–(f): additional ambiguous images obtained after applying the accretive completion algorithm.

6 Discussion

We introduced LABEL classifiers, which are the least ambiguous set-valued classifiers that guarantee certain prediction confidence levels. This framework for classification builds on the strengths of traditional single-valued classifiers, but provides a more informative and principled approach when dealing with ambiguous instances. As we have seen, optimal classifiers — having the smallest expected size — can sometimes output prediction sets that are empty, but we provided several remedies for this problem.

There are many issues that deserve further investigation. Perhaps the most important is how to deal with huge numbers of classes. In this regard, a promising approach is to organize the classes into a structure such as a tree. The tree can be based on prior knowledge or can be discovered from the data.

Finally, we note a possible further enhancement of the method: class discovery. The idea is to look for new classes in the data. For example, if we find well-defined

clusters in a zone where there is either high ambiguity or null predictions, then these observations could potentially correspond to a new class. Similarly, anomalies — observations with low density with respect to each class — could also be considered to define a new class.

A Proofs

We now present the proofs of the results that are not clear from the text.

Lemma 4. Proof. Let \mathbf{H} and \mathbf{H}' be such that

$$\mathbb{P}\{Y \in \mathbf{H}(X)|Y = y\} = \mathbb{P}\{Y \in \mathbf{H}'(X)|Y = y\} = 1 - \alpha_y,$$

and $\mathbb{P}\{y \in \mathbf{H}'(X)|Y \neq y\} \geq \mathbb{P}\{y \in \mathbf{H}(X)|Y \neq y\}$, for all y . Multiplying this expression by $\mathbb{P}(Y \neq y)$ we obtain $\mathbb{P}\{y \in \mathbf{H}'(X), Y \neq y\} \geq \mathbb{P}\{y \in \mathbf{H}(X), Y \neq y\}$, which can be rewritten as

$$\sum_{l \neq y} \mathbb{P}\{y \in \mathbf{H}'(X)|Y = l\} \pi_l \geq \sum_{l \neq y} \mathbb{P}\{y \in \mathbf{H}(X)|Y = l\} \pi_l, \quad (8)$$

which holds for all y . On the other hand we have

$$\sum_y \mathbb{P}\{Y \in \mathbf{H}'(X)|Y = y\} \pi_y = \sum_y \mathbb{P}\{Y \in \mathbf{H}(X)|Y = y\} \pi_y = \sum_y (1 - \alpha_y) \pi_y.$$

Adding Expression (8) over all y and combining with the last expression leads to

$$\sum_y \sum_l \mathbb{P}\{y \in \mathbf{H}'(X)|Y = l\} \pi_l \geq \sum_y \sum_l \mathbb{P}\{y \in \mathbf{H}(X)|Y = l\} \pi_l,$$

which by the law of total probability and Remark 1 is equivalent to $\mathbb{E}\{|\mathbf{H}'(X)|\} \geq \mathbb{E}\{|\mathbf{H}(X)|\}$. \square

Theorem 5. Proof. First, notice that $\text{logit}\{p(y|x)\} = \log\{p(x|y)/p(x|y^c)\} + \text{logit}(\pi_y)$, where $p(x|y^c) \equiv \sum_{j \neq y} p(x|Y = j)\pi_j / \sum_{j \neq y} \pi_j$. Given that the log and logit functions are monotonically increasing, this expression implies that the decision regions C_y can alternatively be based on level sets of the likelihood ratios $\Lambda_y(x) = p(x|y)/p(x|y^c)$, that is $C_y = \{x : \Lambda_y(x) \geq \ell_y\}$ with ℓ_y chosen so that $\mathbb{P}(C_y|Y = y) = 1 - \alpha_y$. The region C_y^c therefore corresponds to the Neyman-Pearson rejection region for testing the null hypothesis $H_0 : Y = y$ versus $H_1 : Y \neq y$. By the Neyman-Pearson lemma we have that the classifier \mathbf{H} induced by the sets C_y maximizes the probabilities $\mathbb{P}\{y \notin \mathbf{H}(X)|Y \neq y\}$, or equivalently $\mathbb{P}\{y \in \mathbf{H}(X)|Y \neq y\}$ is minimized. Finally, by Lemma 4 we have that this decision rule \mathbf{H} also minimizes the ambiguity. \square

Theorem 7. Proof. Firstly, since $\tilde{\mathbb{A}}$ is the optimal value of problem (3), $\tilde{\mathbb{A}} \leq \mathbb{A}(\mathbf{H}^\dagger)$. Now,

$$\begin{aligned} \mathbb{A}(\mathbf{H}^\dagger) &= \mathbb{E} [I\{X \in \mathcal{N}(\mathbf{H}^*)\}|\mathbf{H}^\dagger(X)|] + \mathbb{E} [I\{X \notin \mathcal{N}(\mathbf{H}^*)\}|\mathbf{H}^\dagger(X)|] \\ &= \mathbb{P}\{\mathcal{N}(\mathbf{H}^*)\} + \mathbb{A}(\mathbf{H}^*), \end{aligned}$$

and the result follows from $\mathbb{A}(\mathbf{H}^*) \leq \tilde{\mathbb{A}}$ given that (2) is a relaxation of (3). \square

Theorem 11. Proof. The first part is essentially the same as in Lei (2014). We prove the second part. Let \hat{G}_y be the empirical distribution of $p(y|X_{y,1}), \dots, p(y|X_{y,n_y})$ where $X_{y,1}, \dots, X_{y,n_y}$ are sample points in class y . Let $\hat{\mathbb{P}}_y(\cdot)$ be the probability measure corresponding to \hat{G}_y . Define $L_y(t) = \{x : p(y|x) \leq t\}$, $\hat{L}_y(t) = \{x : \hat{p}(y|x) \leq t\}$.

We focus on the event

$$E = \left\{ \sup_{y,x} |\hat{p}(y|x) - p(y|x)| \leq \epsilon, \sup_{y,t} |\hat{G}_y(t) - G_y(t)| \leq c\sqrt{\frac{\log n}{n}}, \right. \\ \left. \sup_y |\hat{\pi}_y - \pi_y| \leq c\sqrt{\frac{\log n}{n}} \right\},$$

which has probability at least $1 - K\delta - n^{-1}$ if c is chosen large enough and K grows slowly with n . Here the first inequality in E is given by our assumption in (4) and the other two follow from standard empirical process theory.

Recall that for total coverage we use the same threshold for all classes. Let $t^* = G^{-1}(\alpha)$ be the ideal cut-off value for $p(y|x)$. If $t \leq t^* - \epsilon - \{(K+1)cc_1^{-1}\sqrt{\log n/n}\}^{1/\gamma}$, then we have

$$\begin{aligned} \widehat{\mathbb{P}}_y\{\widehat{L}_y(t)\} &\leq \widehat{\mathbb{P}}_y\{L(t+\epsilon)\} = \widehat{G}_y(t+\epsilon) \leq G_y(t+\epsilon) + c\sqrt{\frac{\log n}{n}} \\ &\leq G_y\left[t^* - \{(K+1)cc_1^{-1}\sqrt{\log n/n}\}^{1/\gamma}\right] + c\sqrt{\frac{\log n}{n}} \leq G_y(t^*) - cK\sqrt{\frac{\log n}{n}}. \end{aligned}$$

Therefore,

$$\widehat{t} > t^* - \epsilon - \{(K+1)cc_1^{-1}\sqrt{\log n/n}\}^{1/\gamma}, \quad (9)$$

because otherwise we have

$$\begin{aligned} \sum_{y=1}^K \widehat{\pi}_y \widehat{\mathbb{P}}_y\{\widehat{L}_y(\widehat{t})\} &\leq \sum_{y=1}^K \widehat{\pi}_y \{G_y(t^*) - cK\sqrt{\log n/n}\} \\ &\leq \alpha + \sum_{y=1}^K |\widehat{\pi}_y - \pi_y| G_y(t^*) - cK\sqrt{\log n/n} < \alpha. \end{aligned}$$

Similarly we can obtain

$$\widehat{t} \leq t^* + \epsilon + \{(K+1)cc_1^{-1}\sqrt{\log n/n}\}^{1/\gamma}, \quad (10)$$

and combining (9) and (10) we have $|\widehat{t} - t^*| \leq \epsilon + \{(K+1)cc_1^{-1}\sqrt{\log n/n}\}^{1/\gamma}$. (It is worth noting that a rigorous argument of this would require $\widehat{p}(y|x)$ to have distinct values at the sample points X_1, \dots, X_n . This is a minor issue because one can always add very small random perturbations such as $\widehat{p}(y|X) + \xi$ with $\xi \sim \text{Unif}(-n^{-2}, n^{-2})$.)

Then

$$\begin{aligned}
\mathbb{P}_y \left(\widehat{C}_y \setminus C_y^* \right) &= \mathbb{P}_y \left\{ \widehat{p}(y|X) \geq \widehat{t}, p(y|X) < t^* \right\} \\
&\leq \mathbb{P}_y \left[t^* - 2\epsilon - \left\{ (K+1)cc_1^{-1} \sqrt{\log n/n} \right\}^{1/\gamma} \leq p(y|X) < t^* \right] \\
&\leq c' \left(\epsilon^\gamma + K \sqrt{\log n/n} \right),
\end{aligned}$$

for some constant c' depending on c, c_1, γ . Similarly we can obtain $\mathbb{P}_y(C_y^* \setminus \widehat{C}_y) \leq c'(\epsilon^\gamma + K \sqrt{\log n/n})$, and hence $\mathbb{P}_y(\widehat{C}_y \triangle C_y^*) \leq c'(\epsilon^\gamma + K \sqrt{\log n/n})$. Summing over y we have

$$\mathbb{P} \left(\widehat{\mathbf{H}} \triangle \mathbf{H}^* \right) = \sum_{y=1}^K \pi_y \mathbb{P}_y(\widehat{C}_y \triangle C_y^*) \leq c' \left(\epsilon^\gamma + K \sqrt{\log n/n} \right).$$

□

References

- Audibert, J.-Y., & Tsybakov, A. B. (2007). Fast learning rates for plug-in classifiers. *The Annals of Statistics*, 35, 608–633.
- Bartlett, P. L., & Wegkamp, M. H. (2008). Classification with a reject option using a hinge loss. *The Journal of Machine Learning Research*, 9, 1823–1840.
- Chow, C. K. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16, 41–46.
- del Coz, J. J., Diez, J., & Bahamonde, A. (2009). Learning nondeterministic classifiers. *The Journal of Machine Learning Research*, 10, 2273–2293.
- Denis, C., & Hebiri, M. (2015). Consistency of plug-in confidence sets for classification in semi-supervised learning. *arXiv:1507.07235*.

- Devroye, L. (1978). The uniform convergence of nearest neighbor regression function estimators and their application in optimization. *IEEE Transactions on Information Theory*, 24(2), 142–151.
- Grycko, E. (1993). Classification with set-valued decision functions. In O. Opitz, B. Lausen, & R. Klar (Eds.) *Information and Classification*, Studies in Classification, Data Analysis and Knowledge Organization, (pp. 218–224). Springer Berlin Heidelberg.
- Herbei, R., & Wegkamp, M. H. (2006). Classification with Reject Option. *Canadian Journal of Statistics*, 34(4), 709–721.
- Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In D. S. Touretzky (Ed.) *Advances in Neural Information Processing Systems 2. Proceedings of the 1989 Conference*, (pp. 396–404). Morgan Kaufmann.
- Lei, J. (2014). Classification with confidence. *Biometrika*, 101(4), 755–769.
- Lei, J., Rinaldo, A., & Wasserman, L. (2014). A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, (p. in press).
- Lei, J., Robins, J., & Wasserman, L. (2013). Distribution free prediction set. *Journal of the American Statistical Association*, 108, 278–287.
- Lei, J., & Wasserman, L. (2014). Distribution free prediction bands for nonparametric regression. *Journal of the Royal Statistical Society, Series B*, 76, 71–96.
- Lichman, M. (2013). UCI machine learning repository.
URL <http://archive.ics.uci.edu/ml>

- Papadopoulos, H., Proedrou, K., Vovk, V., & Gammerman, A. (2002). Inductive confidence machines for regression. In *Machine Learning: ECML 2002*, (pp. 345–356). Springer.
- Ramaswamy, H. G., Tewari, A., & Agarwal, S. (2015). Consistent algorithms for multiclass classification with a reject option. *arXiv preprint arXiv:1505.04137*.
- Shafer, G., & Vovk, V. (2008). A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9, 371–421.
- Silverman, B. W. (1986). *Density estimation for statistics and data analysis*, vol. 26. CRC press.
- Stone, C. (1982). Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, 10, 1040–1053.
- Tsybakov, A. B. (2009). *Introduction to Nonparametric Estimation*. New York: Springer.
- van de Geer, S. A. (2008). High-dimensional generalized linear models and the lasso. *The Annals of Statistics*, 36, 614–645.
- Vovk, V. (2013). Conditional validity of inductive conformal predictors. *Machine Learning*, 92, 349–376.
- Vovk, V., Gammerman, A., & Shafer, G. (2005). *Algorithmic Learning in a Random World*. New York: Springer.
- Yuan, M., & Wegkamp, M. (2010). Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research*, 11, 111–130.